Agent-based modelling of market competition among flexibility options using machine-learning techniques

Felix Nitsch[1,*], Christoph Schimeczek[1], Valentin Bertsch[2]

[1] Department of Energy Systems Analysis
Institute of Networked Energy Systems
German Aerospace Center (DLR)
Curiestraße 4, 70563 Stuttgart, Germany
* corresponding author: Felix.Nitsch@dlr.de +49 711 6862-8865

[2] Chair of Energy Systems & Energy Economics
Ruhr-Universität Bochum
Universitätsstraße 150, 44801 Bochum, Germany

PARIS AGREEMENT – THE ENERGY TRANSITION IS SET

Credits: BMU/Sascha Hilgers

to be open source in 2021

Simulating electricity markets with AMIRIS

Agent-based model AMIRIS developed at DLR Stuttgart (Deissenroth et al., 2017) simulating (German) electricity markets

Input

- RE feed-in
- Load
- Power plant park
- Efficiencies
- Availabilities
- Fuel costs
- CO_2 costs

Output

- Electricity prices
- Power plant dispatch
- Storage dispatch
- Market values
- Emissions
- System costs

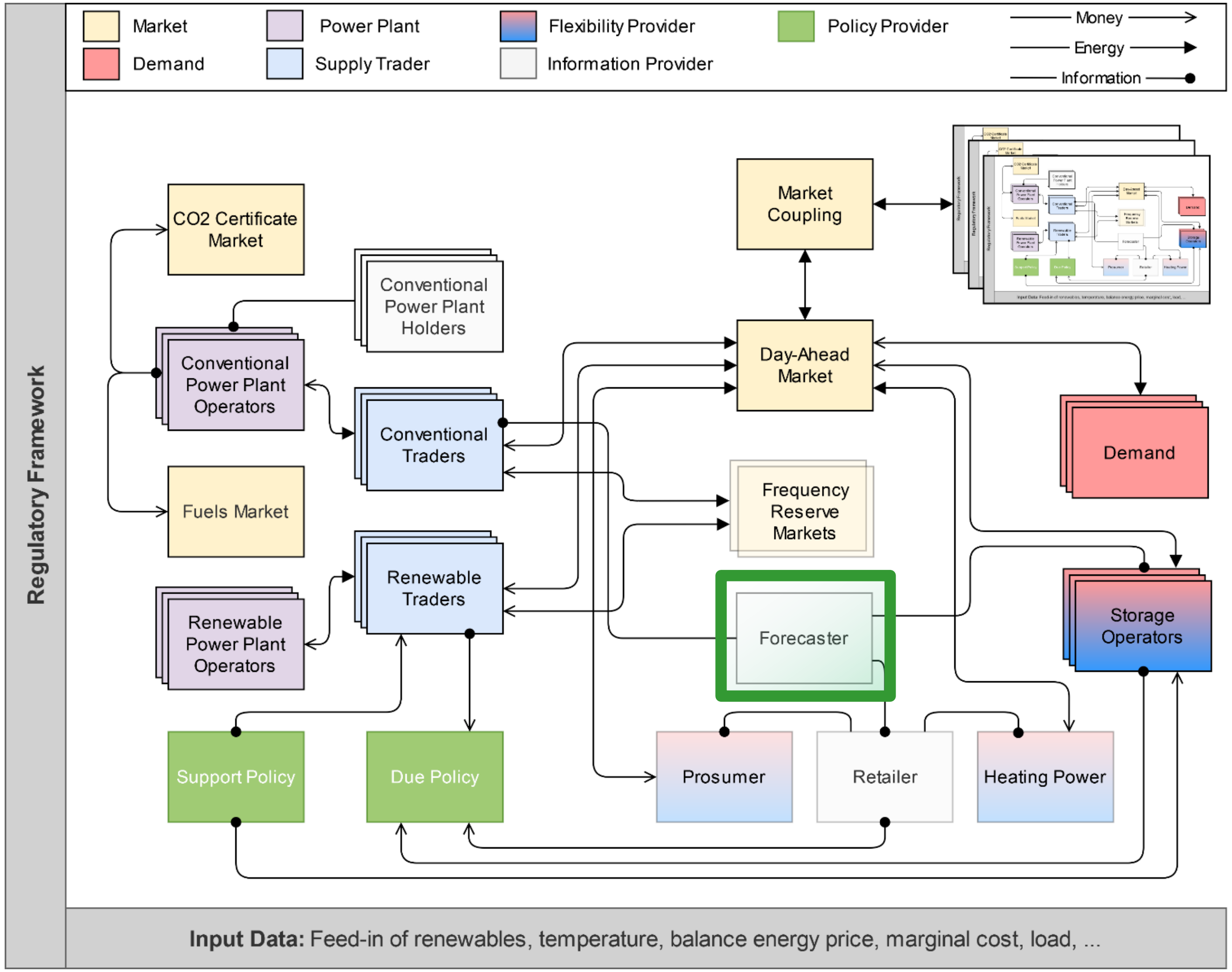


Fig. 1: Model setup of AMIRIS

Status Quo: Price forecasting in AMIRIS

1. All power plant operators send their bids through respective traders to forecast agent in advance
2. The forecast agent calculates preliminary merit-order resulting in forecasted price
3. Forecasted prices is sent to flexibility option agent
4. The flexibility option agent optimizes its operational strategy
5. All traders (incl. Flexibility options) send their final bids to Electricity Exchange
6. The market clearing reveals final electricity price which may deviate based on operation of flexibility option agent action *(e.g. charging → „higher price“, discharging → „lower price“)*

Challenge:

Multiple flexibility option agents may distort this „simple“ forecast due to their competitive actions

→ Significant impacts on the accuracy of the price forecast

Preliminary work

Aim:

Central forecast agent is learning bidding behaviour of flexibility options and their impacts on prices

Architecture:

- 1. Feed-forward model (FF)
- 2. Long-short term memory model (LSTM)
- Inputs:
  - Previous prices
  - Previous residual load
- Output:
  - Forecast for next 3 hours

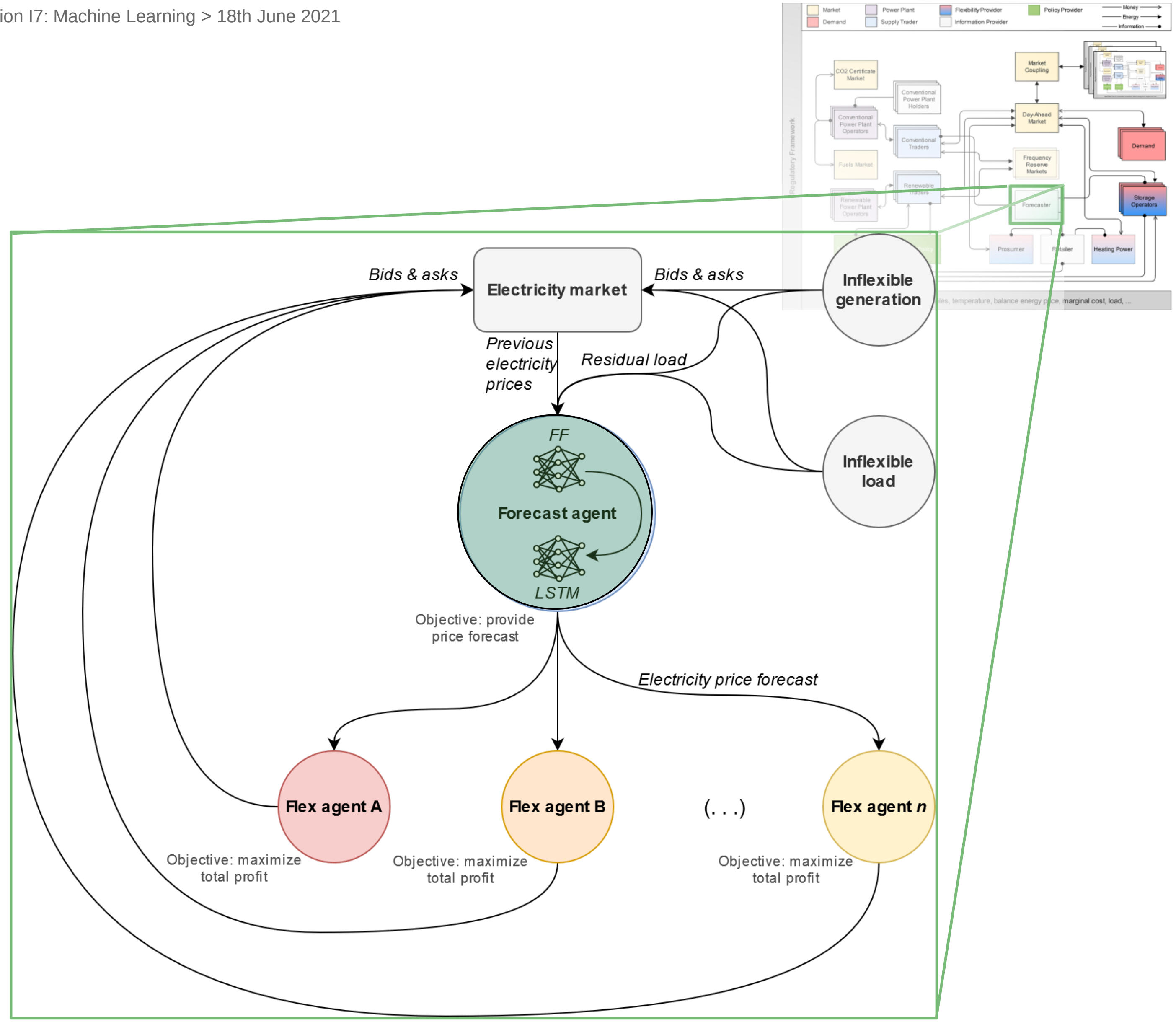


Fig.2: Forecast agent equipped with neural networks providing forecasts for multiple flexibility options

Preliminary work:
Predicted prices with FF and LSTM model

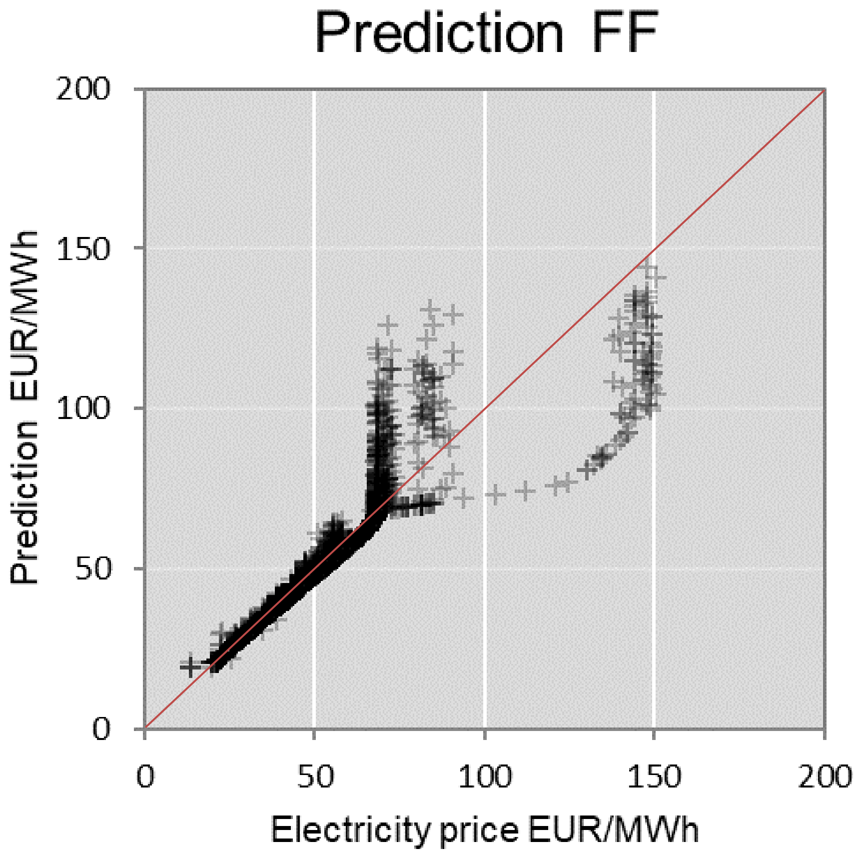


Fig.3: Predicted prices against simulated prices from FF network

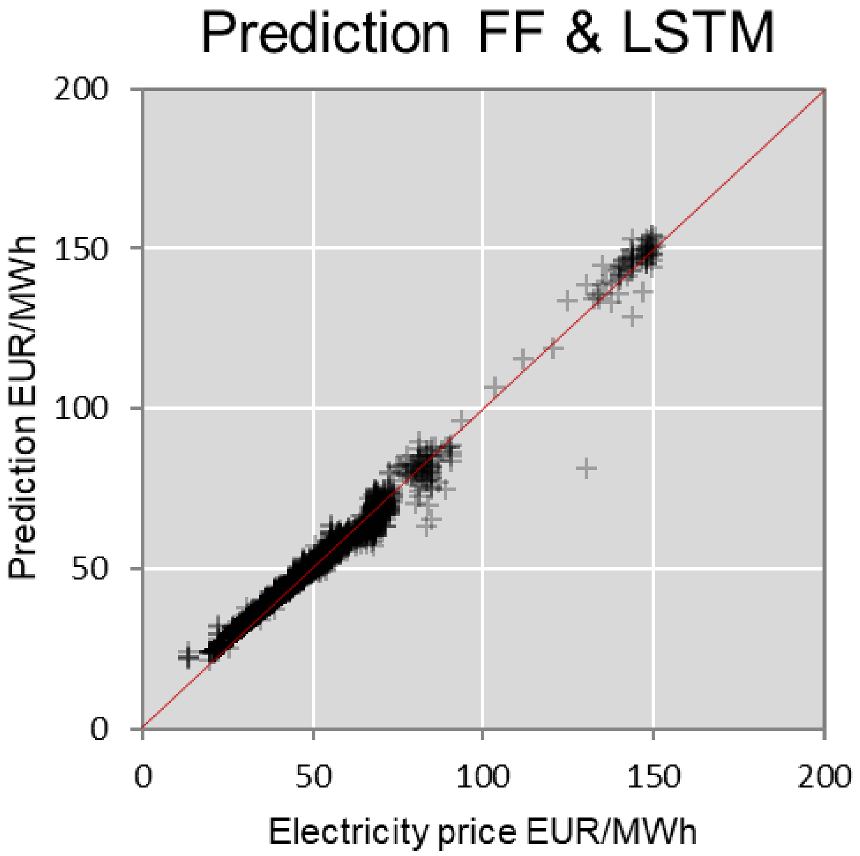


Fig.4: Predicted prices against simulated prices from LSTM network using FF predictions and simulated prices as input

Nitsch, F. and Schimeczek C. (2020). Model in model: Electricity price forecasts in agent-based energy system simulations. INREC Conference. https://elib.dlr.de/136017/

DLR

Open challenges

1. Accurate predictions only for t+1, t+2, and t+3 time steps
    - *Sufficient for proof-of-concept, yet too little information for building operational strategy in real application*

2. No way to consider uncertainties regarding the prediction values
    - *Does not allow different levels of risk-aversion/affinity regarding the optimal operational strategy*

3. Inconvenient two-staged training process (1. FF → 2. LSTM)
    - *Requires many adaptations regarding the consideration of different time horizons*

4. “Black box" characteristic of ML prediction
    - *Trial and error of which variables are crucial to improve results*

Temporal Fusion Transformers

- Novel attention-based architecture proposed by Lim et al. (2020)
- Significant performance improvements over existing benchmarks (see Table)

- <u>Main features:</u>
  - Gating mechanisms to skip over any unused components
  - Variable selection networks to select relevant input variables
  - Static covariate encoders to integrate static features
  - Temporal processing to learn both long- and short-term temporal relationships from both observed and known time-varying inputs

Table 2: P50 and P90 quantile losses on a range of real-world datasets. Percentages in brackets reflect the increase in quantile loss versus TFT (lower q-Risk better), with TFT outperforming competing methods across all experiments, improving on the next best alternative method (<u>underlined</u>) between 3% and 26%.

	ARIMA	ETS	TRMF	DeepAR	DSSM
Electricity	0.154 (*+180%*)	0.102 (*+85%*)	0.084 (*+53%*)	0.075 (*+36%*)	0.083 (*+51%*)
Traffic	0.223 (*+135%*)	0.236 (*+148%*)	0.186 (*+96%*)	0.161 (*+69%*)	0.167 (*+76%*)
	ConvTrans	**Seq2Seq**	**MQRNN**	**TFT**	
Electricity	0.059 <u>(*+7%*)</u>	0.067 (*+22%*)	0.077 (*+40%*)	**0.055***	
Traffic	0.122 (*+28%*)	0.105 <u>(*+11%*)</u>	0.117 (*+23%*)	**0.095***	

(a) P50 losses on simpler univariate datasets.

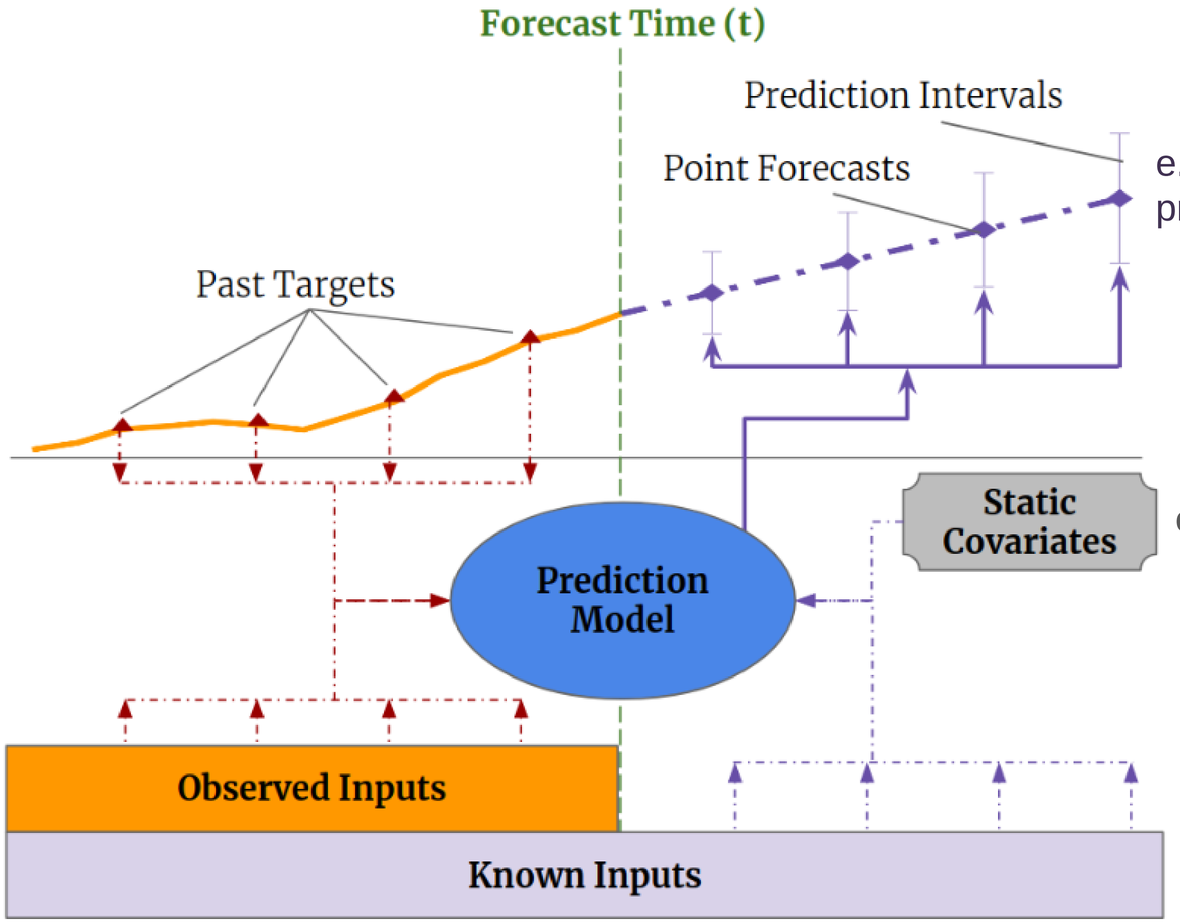
	ARIMA	ETS	TRMF	DeepAR	DSSM
Electricity	0.102 (*+278%*)	0.077 (*+185%*)	-	0.040 (*+48%*)	0.056 (*+107%*)
Traffic	0.137 (*+94%*)	0.148 (*+110%*)	-	0.099 (*+40%*)	0.113 (*+60%*)
	ConvTrans	**Seq2Seq**	**MQRNN**	**TFT**	
Electricity	0.034 <u>(*+26%*)</u>	0.036 (*+33%*)	0.036 (*+33%*)	**0.027***	
Traffic	0.081 (*+15%*)	0.075 <u>(*+6%*)</u>	0.082 (*+16%*)	**0.070***	

(b) P90 losses on simpler univariate datasets.

Lim, B., Arik, S. O., Loeff, N., & Pfister, T. (2020). Temporal fusion transformers for interpretable multi-horizon time series forecasting. arXiv preprint arXiv:1912.09363.

Concept of forecasting using TFT

i.e. present time



e.g. next 24 hours including pre-defined quantiles

e.g. previous 24 hours which where observed

e.g. region

e.g. prices, load

e.g. load forecast, calendar information

Fig.5: Inputs and Outputs of TFT model, adapted from Lim et al. (2020)

TFT Architecture

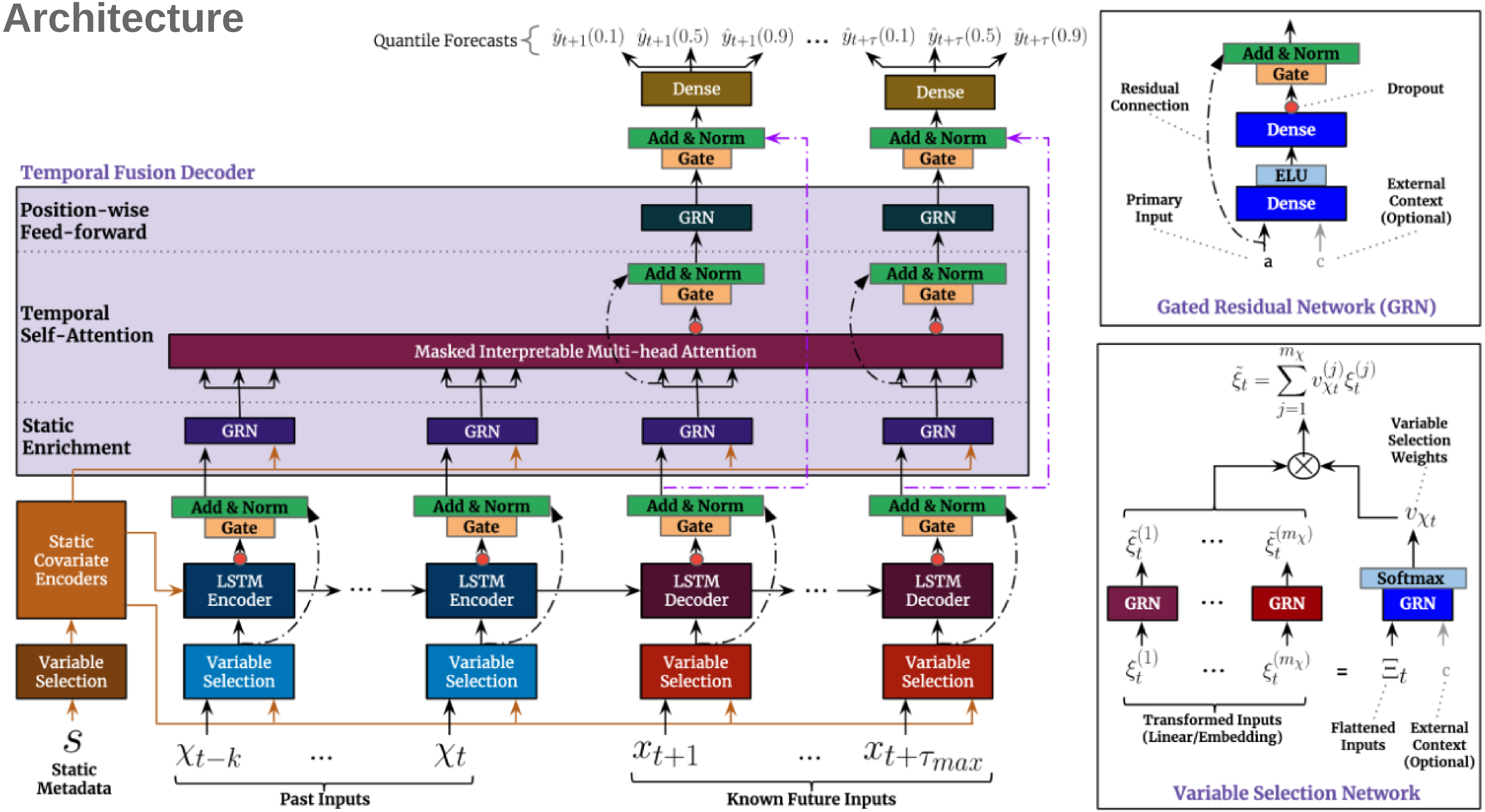


Fig.6: Illustration of TFT architecture, Lim et al. (2020)

Workflow

- **1. Hyperparameter scan**
  - Number of hidden layers
  - Dropout
  - Learning rate
  - Gradient clip value
  - etc.

- **2. Training**
  - Use same data as in Nitsch et al. (2020)
    - Scenario I: no flexibility options (easy to forecast, electric load ⇔ price)
    - Scenario II: extensive flexibility options (challenging to forecast, electric load ⇎ price)

- **3. Analysis**
  - Prediction curves
  - Attention plots

Results: simulated prices in scenario without flexibility options

Encoder 24h – Prediction 6h (v_33) and 24h (v_34)

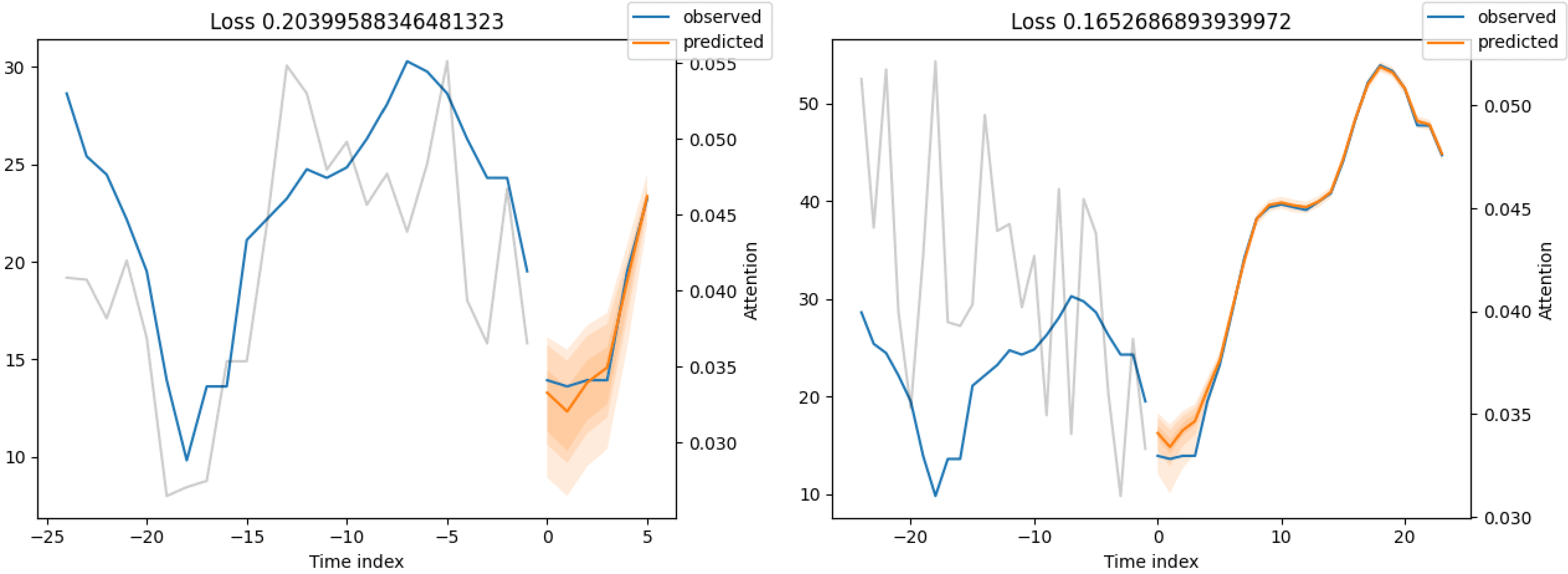
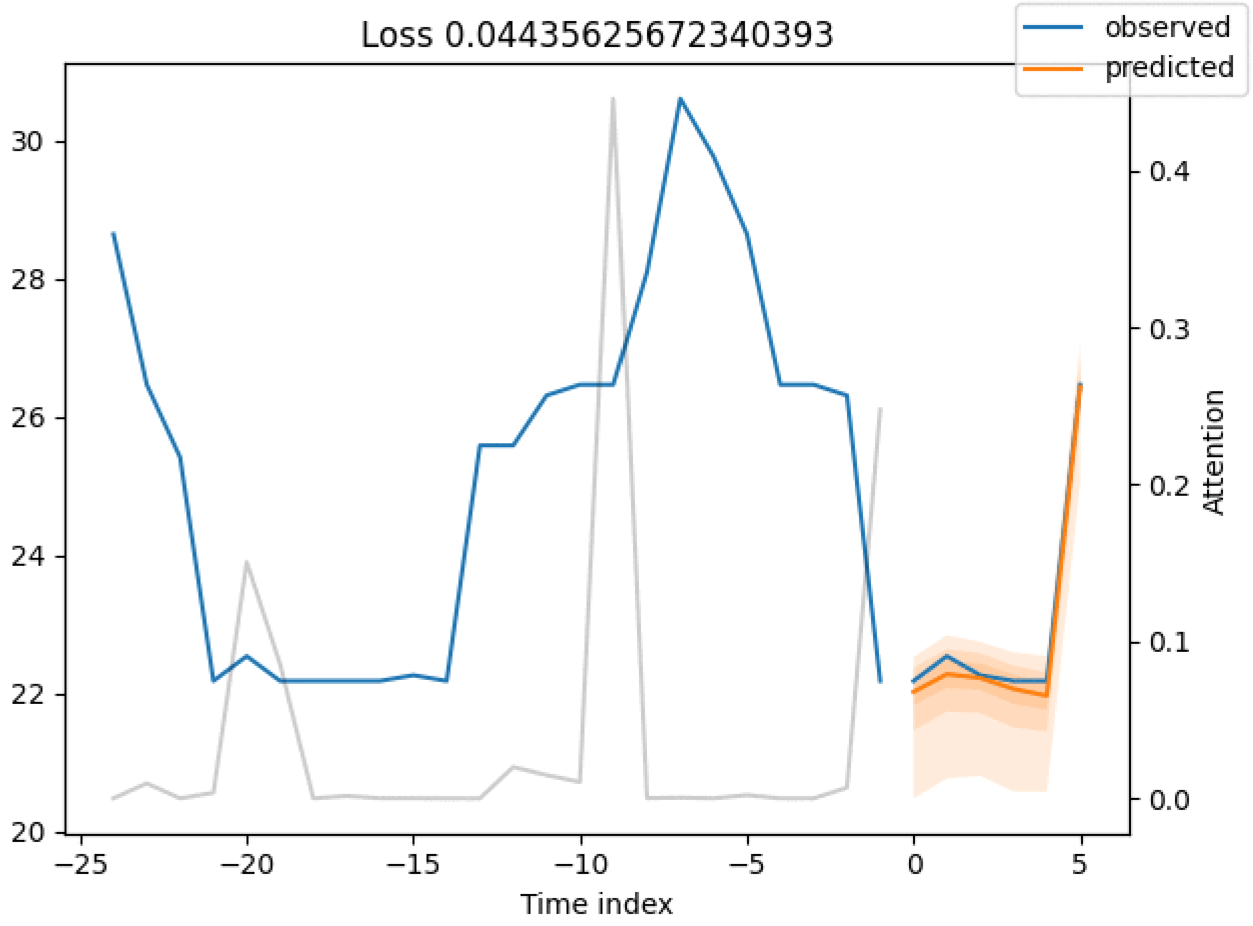


Fig.7: Plotting of observed prices (blue) and predictions (orange) including quantiles [0.02, 0.1, 0.25, 0.5, 0.75, 0.9, 0.98] in EUR/MWh in Scenario I

Results: simulated prices in extensive flexibility option scenario

Encoder 24h – Prediction 6h (v_35) and 24h (v_36)



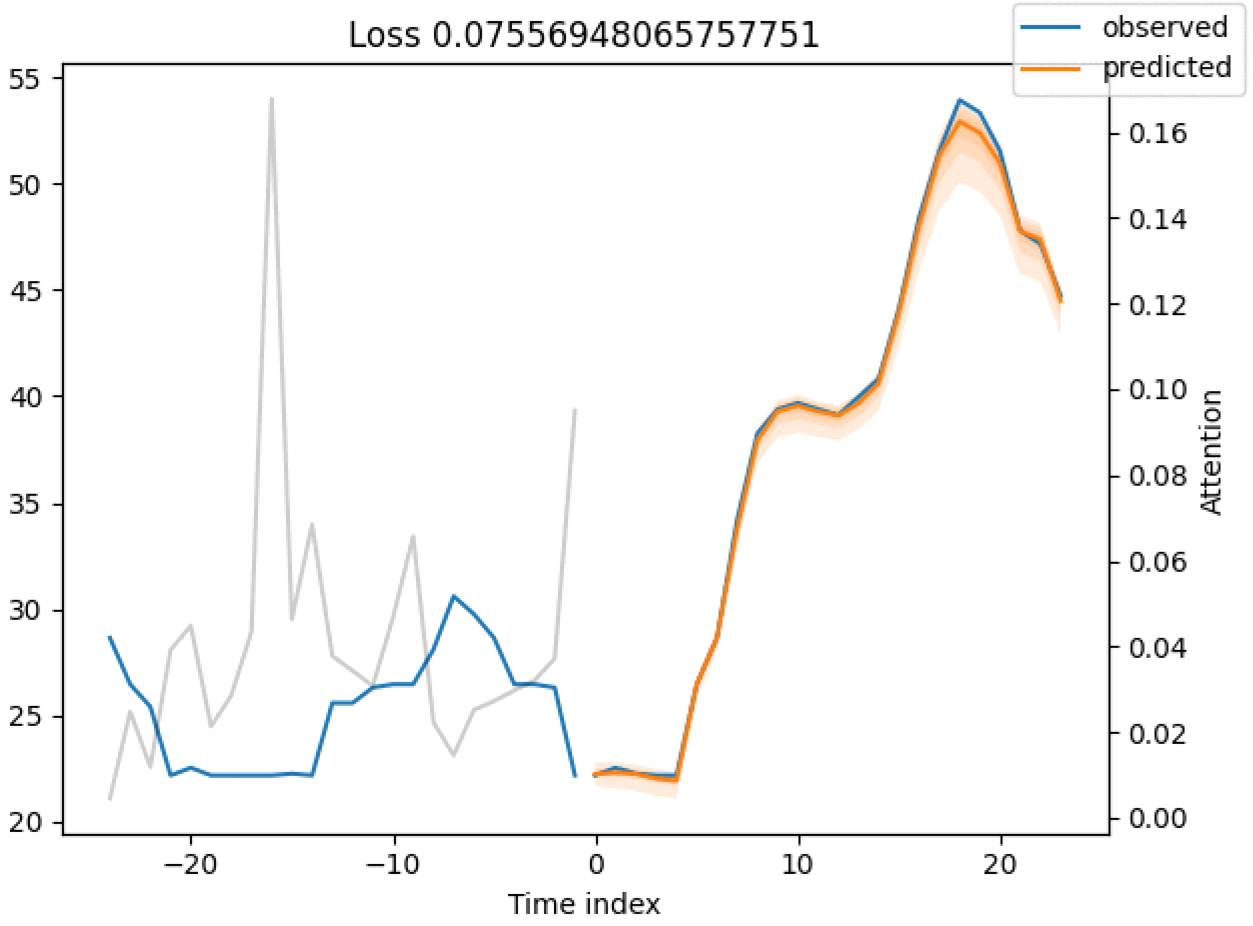


Fig.8: Plotting of observed prices (blue) and predictions (orange) including quantiles [0.02, 0.1, 0.25, 0.5, 0.75, 0.9, 0.98] in EUR/MWh in Scenario II

Results: „time step attention“ in extensive flexibility option scenario

Encoder 24h – Prediction 6h (v_35) and 24h (v_36)

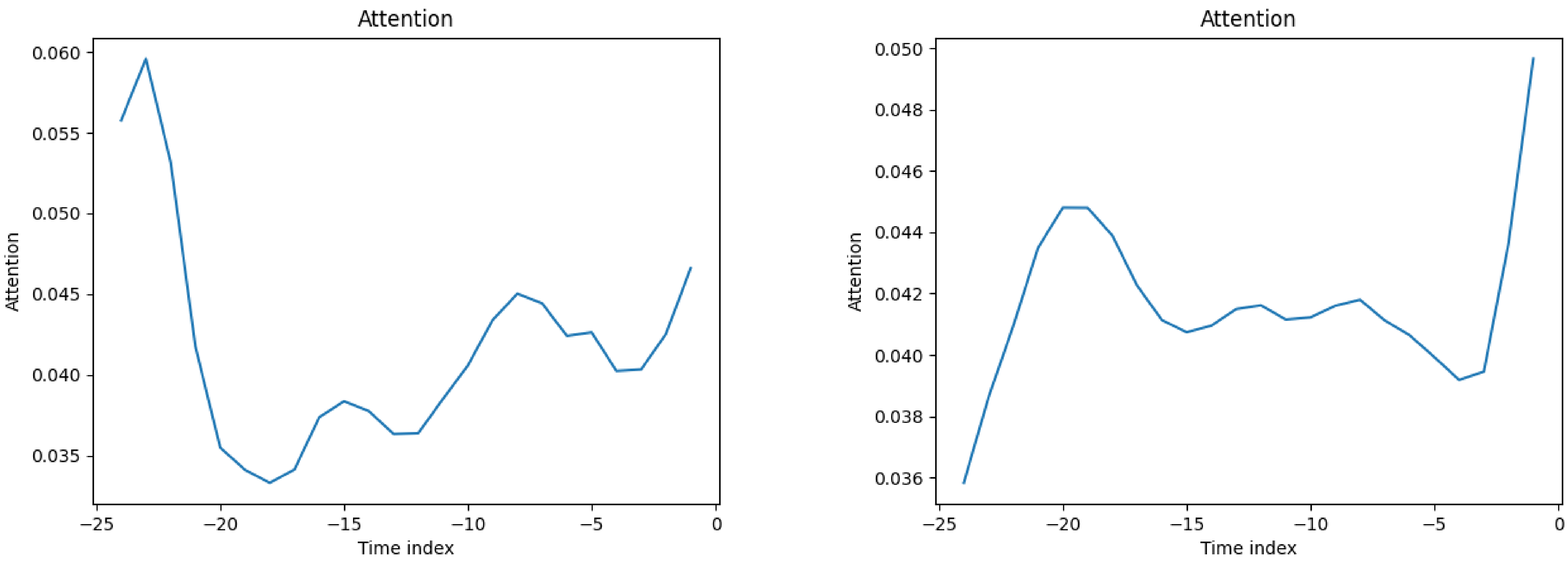


Fig.9: Attention on time steps in TFT model indicating relative importance of each observed time step

Results: variable importance in extensive flexibility option scenario

Encoder 24h – Prediction 6h (v_35) and 24h (v_36)

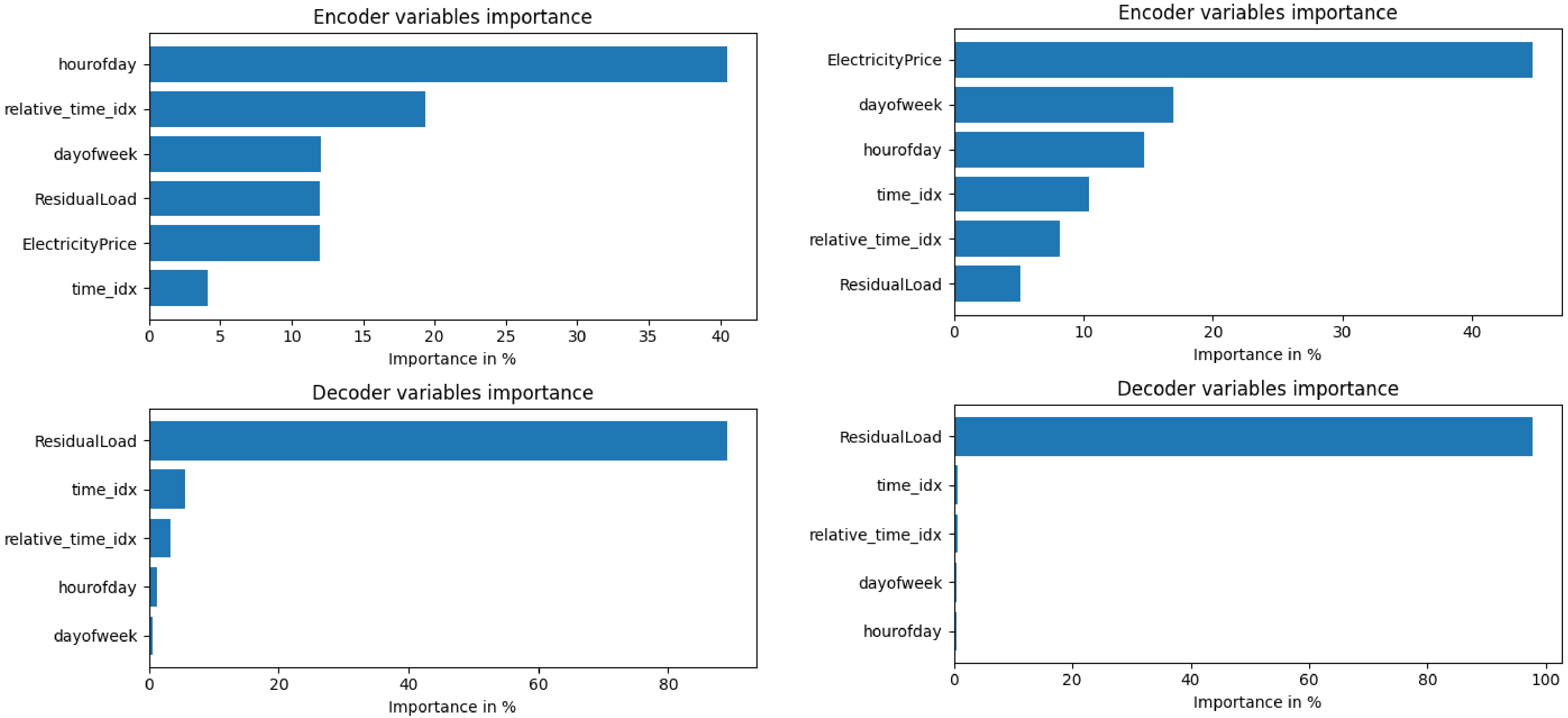


Fig.10: Attention of variables in TFT model for encoder variables (observed values in past) and decoder variables (known values in future)

Excursus: Applying TFT to real-world electricity prices

- Data source:
  Open Power System Data. 2020. *Data Package Time series.* Version 2020-10-06.
  https://doi.org/10.25832/time_series/2020-10-06

- Values on:
  - load forecast
  - actual load
  - wind generation
  - solar generation
  - historic prices
  - calendar information

Excursus: Applying TFT to real-world electricity prices

Encoder 24h – Prediction 6h (v_30) and Encoder 72h – Prediction 6h (v_31)

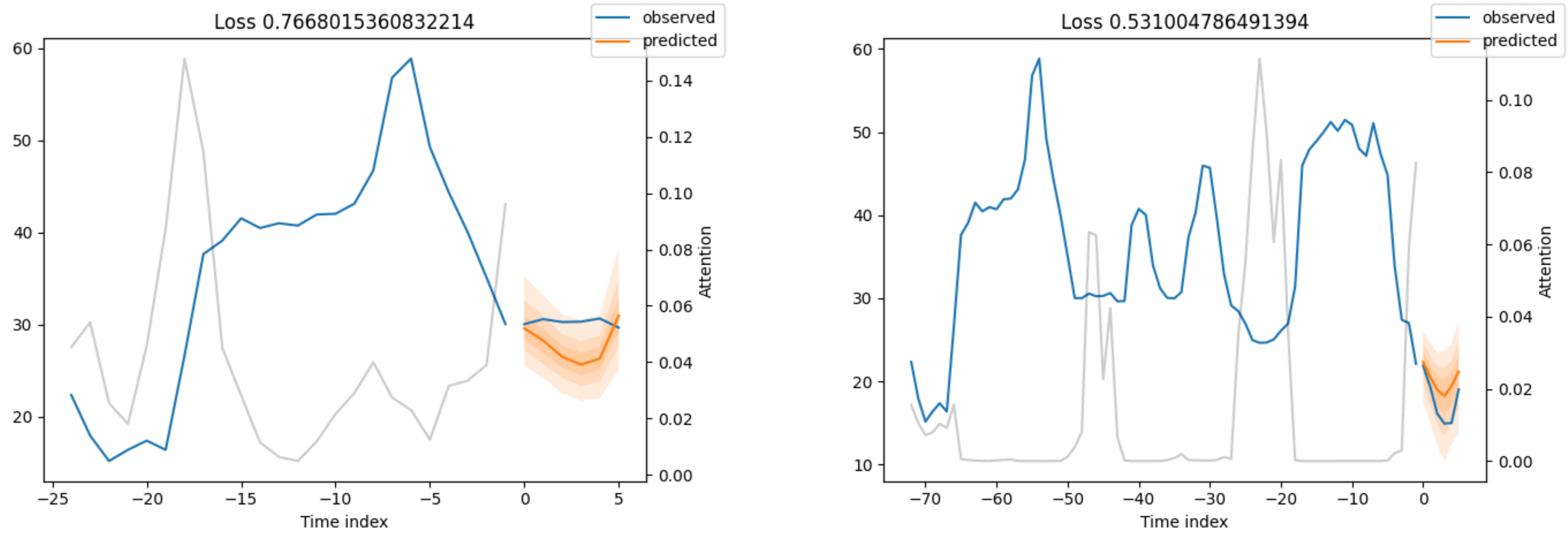


Fig.11: Plotting of observed prices (blue) and predictions (orange) including quantiles [0.02, 0.1, 0.25, 0.5, 0.75, 0.9, 0.98] in EUR/MWh for real world data

Conclusion

1. ~~Accurate predictions only for t+1, t+2, and t+3 time steps~~
   ► High quality forecasts for 24 time steps (MAE: 0.47 EUR/MWh)

2. ~~No way to consider uncertainties regarding the prediction values~~
   ► Prediction uncertainty estimates provided by TFT architecture

3. ~~Inconvenient two-staged training process (1. FF → 2. LSTM)~~
   ► Very convenient training using pytorch and TFT implementation by Jan Beitner

4. ~~“Black box" characteristic of ML prediction~~
   ► “Attention" feature: identify time steps and input variables relevant for good predictions

Outlook

- Generalize training data for scenarios
- Interface price forecasting ML models (Python) in market simulation AMIRIS (Java)

Contact: Felix Nitsch felix.nitsch@dlr.de German Aerospace Center | Institute of Networked Energy Systems | Energy Systems Analysis